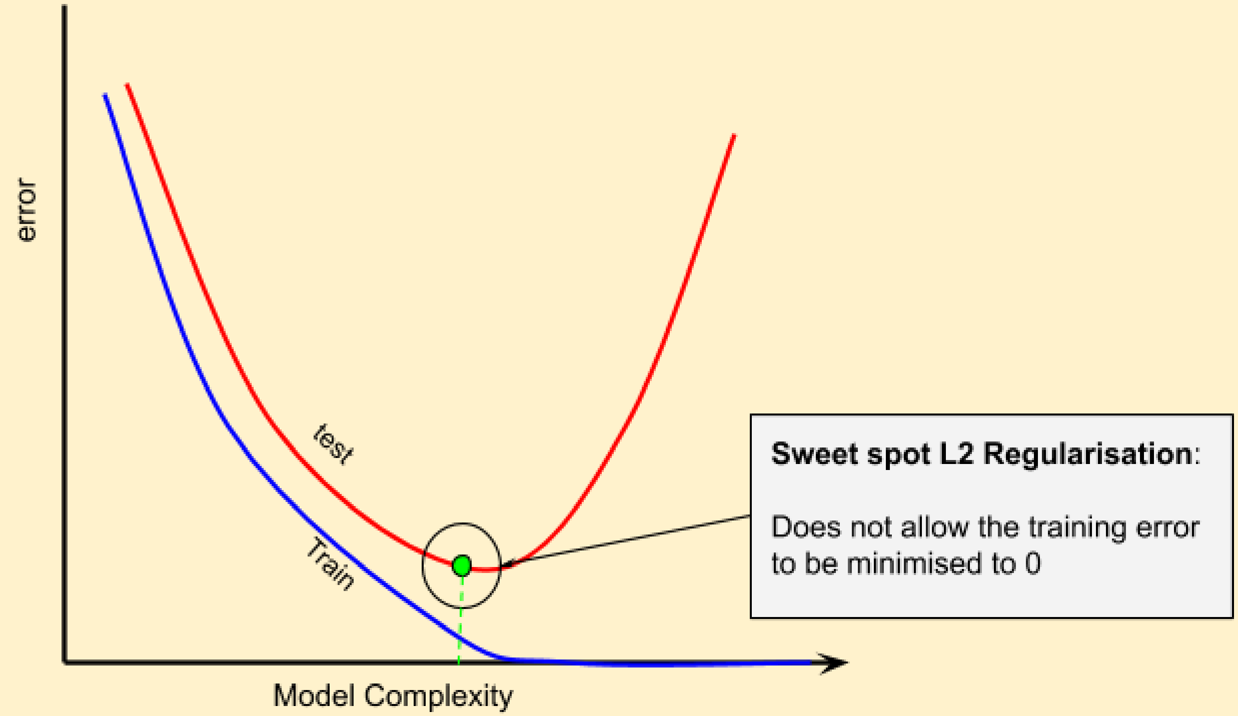
PadhAI: Regularization

One Fourth Labs

L2 regularization

What is the intuition behind L-2 regularization?

1. Consider the error curves for training and test set



2. In the case of Square error loss: $L_{train}(\theta) = \sum_{i=1}^{N}(y_i - \hat{f}(x_i))^2$
    a. Where $\theta = [W_{111},\ W_{112},\ + \ldots + W_{Lnk}]$
    b. Our aim has been to minimise the loss function $min_\theta\ L(\theta)$
3. Now, imagine if we include a new term in the minimization condition $min_\theta\ L(\theta) = L_{train}(\theta) + \Omega(\theta)$
    a. Here, in addition to minimising the training loss, we are also minimising some other quantity that is dependent on our parameters
    b. In the case of L2 Regularisation, $\Omega(\theta) = ||\theta||^2{}_2$ (sq.root of the sum of the squares of the weight)
    c. $\Omega(\theta) = W^2{}_{111} + W^2{}_{112} + \ldots + W^2{}_{Lnk}$
    d. Here, we should aim to minimize both $L_{train}(\theta)\ and\ \Omega(\theta)$, it wouldn't make sense for either of them to be high values.
4. What if we set all weights to 0? In this case, the model would not have learned much, therefore $L_{train}(\theta)$ would be high.
5. What if we try to minimise $L_{train}(\theta)$ to 0? In this case, it is possible that some of the weights would take on large values, thereby driving the value of $\Omega(\theta)$ high.
6. To counter the previous point's shortcoming, we need to minimize $L_{train}(\theta)$ but shouldn't allow the weights to grow too large
7. Thus, as shown in the figure, in L2 Regularisation, we do not allow the training loss to be brought to be zero, instead we maintain it at slightly above zero, so that $\Omega(\theta)$ doesn't become too high
8. This works in the Gradient Descent Algorithm as well

---

9. The algorithm
   a. **Initialise:** w_{111}, w_{112}, ... w_{313}, b_1, b_2, b_3 randomly
   b. **Iterate over data**
      i. Compute ŷ
      ii. Compute L(w,b) Cross-entropy loss function
      iii. $w_{111} = w_{111} - \eta \Delta w_{111}$
      iv. $w_{112} = w_{112} - \eta \Delta w_{112}$
      ...
      v. $w_{313} = w_{111} - \eta \Delta w_{313}$
   c. **Till satisfied**
10. The derivative of the loss function w.r.t any weight is $\Delta W_{ijk} = \frac{\partial L(\theta)}{\partial W_{ijk}}$
11. In the case of L2 Regularisation, that value would be $\Delta W_{ijk} = \frac{\partial L_{train}(\theta)}{\partial W_{ijk}} + \frac{\partial \Omega(\theta)}{\partial W_{ijk}}$
12. Here, the derivative of the regularisation term will cancel out all other weights except the concerned weight and we will compute its derivative. I.e. $\frac{\partial \Omega(\theta)}{\partial W_{ijk}} = 2W_{ijk}$
13. So the new derivative term will be $\Delta W_{ijk} = \frac{\partial L_{train}(\theta)}{\partial W_{ijk}} + 2W_{ijk}$
14. This process is automatically done in PyTorch.